

Fig. 1 Prior art

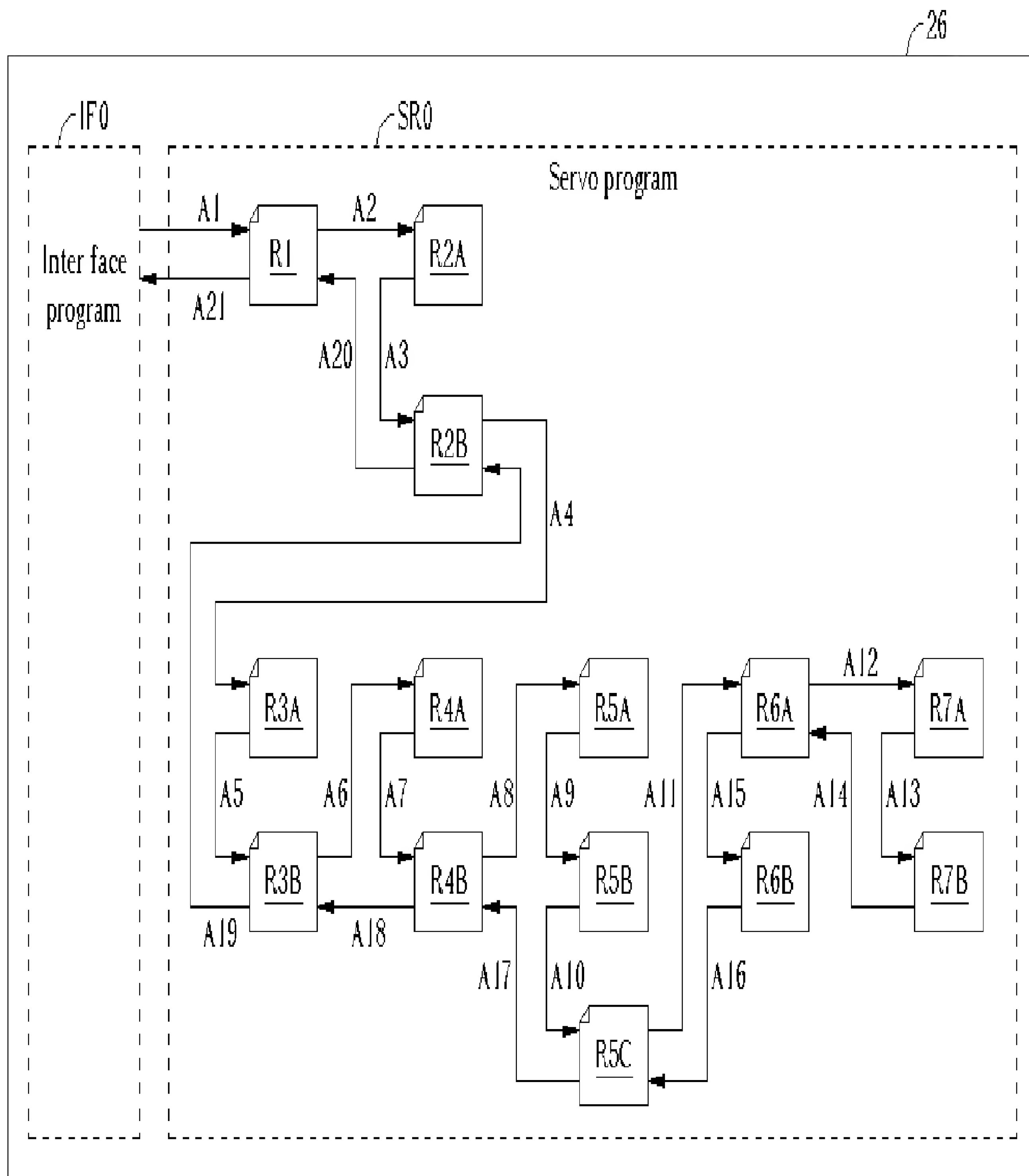


Fig. 2 Prior art

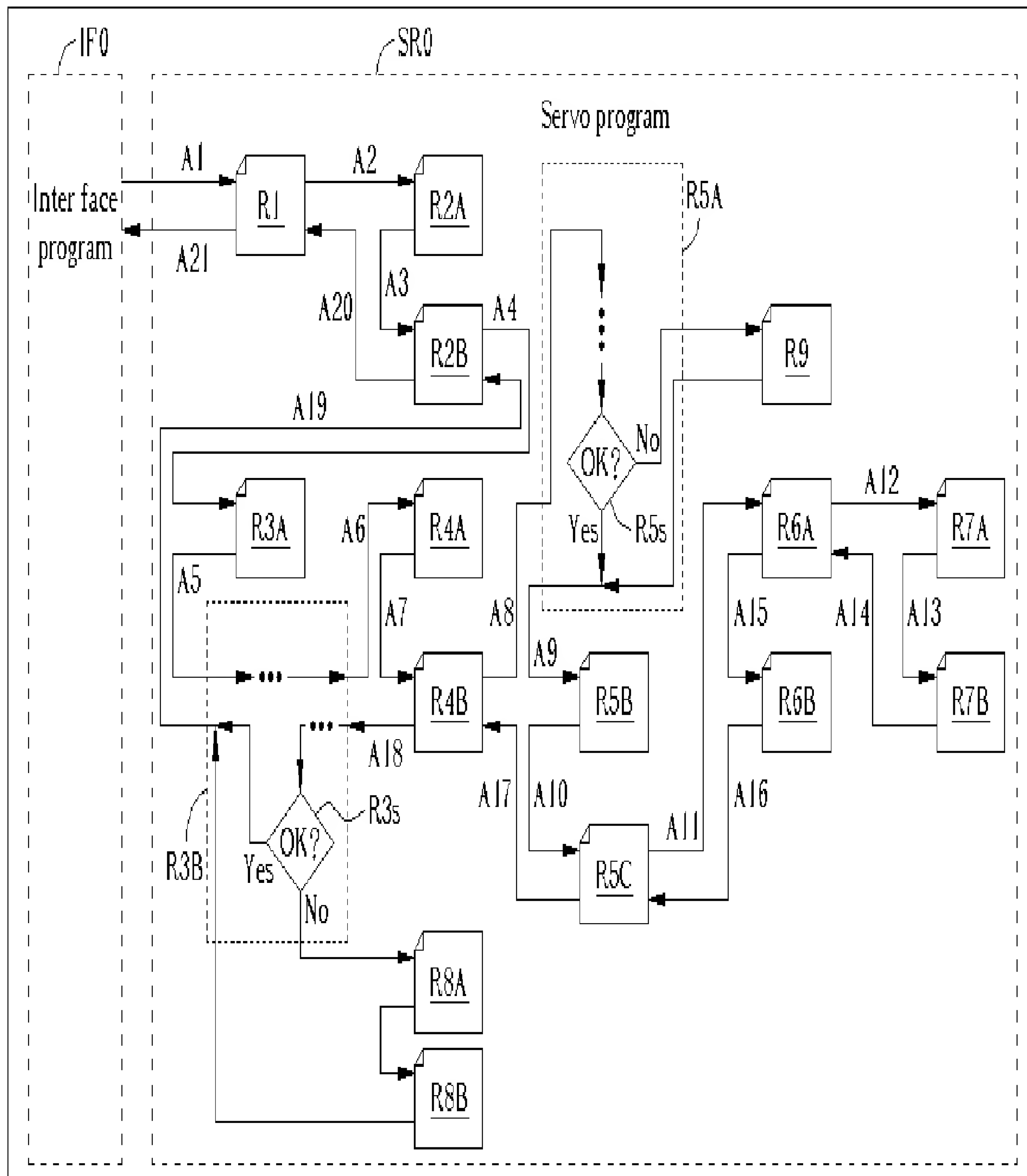


Fig. 3 Prior art

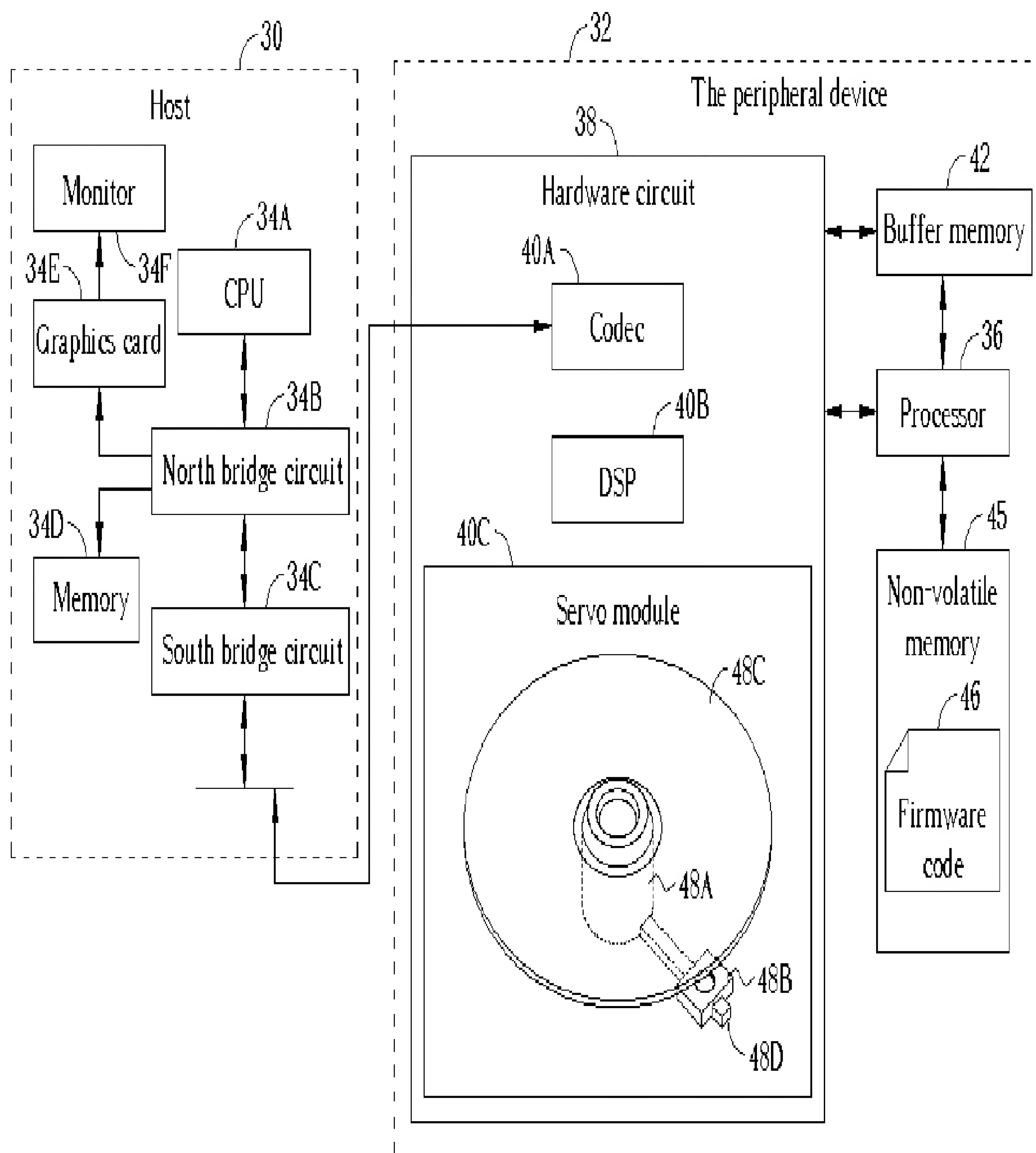


Fig. 4

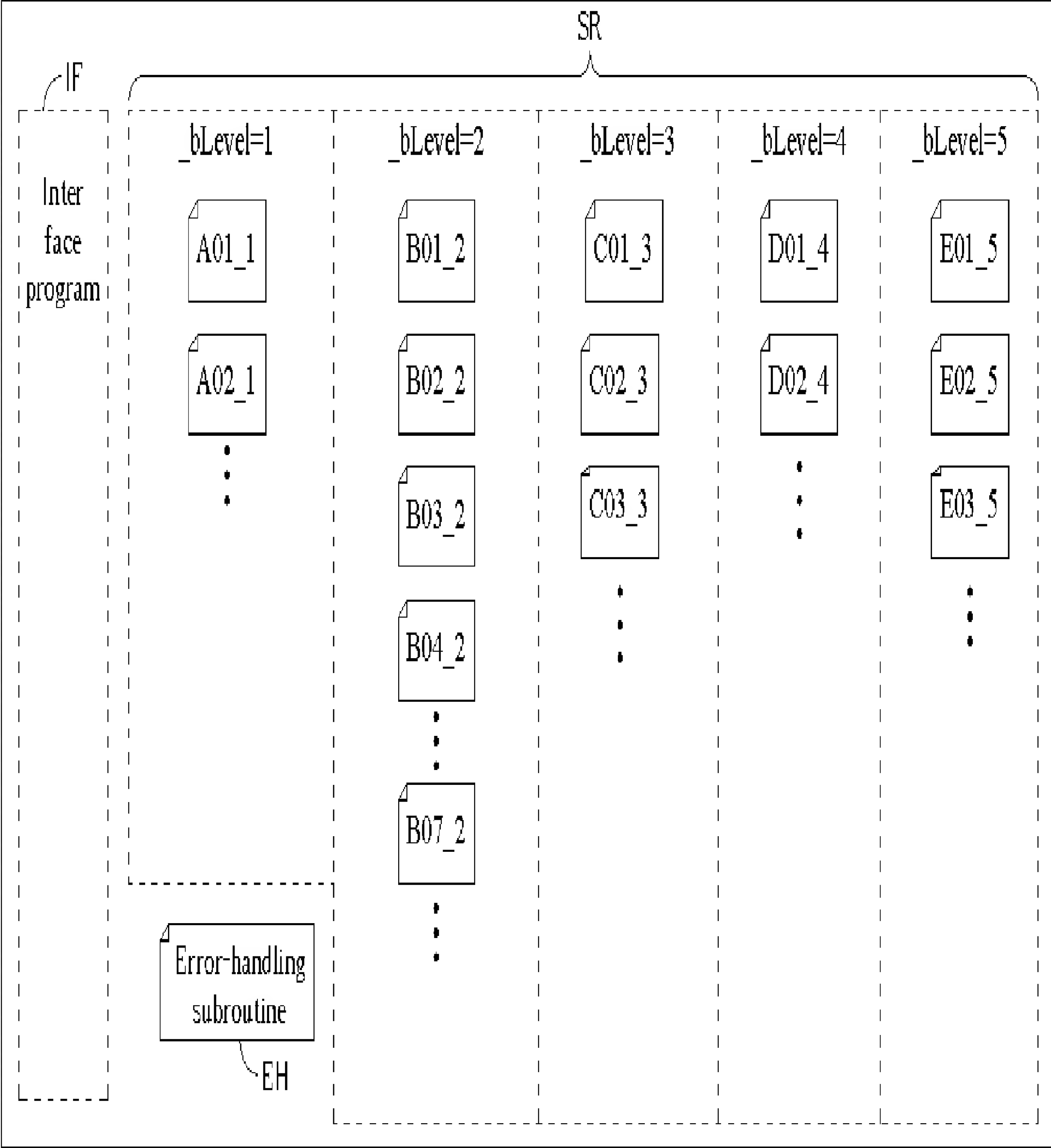


Fig. 5

```

#define ChkStatus(x)  x
#define SetStatus(x)  x = 1;
#define ClrStatus(x)  x = 0;

BYTE A01_1(...)
{
    ++_bLevel;
    ...
    /* Global status check */
    if (ChkStatus( _fgSelectB01_2))
    {
        if (B01_2(...) != READY)
        {
            _bErrorCode[ _bLevel--] = B01_Err;
            return(!READY);
        }
    }
    else
    {
        if (B02_2(...) != READY)
        {
            _bErrorCode[ _bLevel--] = B02_Err;
            return(!READY);
        }
    }
    ...
}

...
50A {
    if (B03_2(...) != READY)
    {
        _bErrorCode[ _bLevel--] = B03_Err;
        return(!READY);
    }
    ...
}

50B {
    if (B04_2(...) != READY)
    {
        _bErrorCode[ _bLevel--] = B04_Err;
        return(!READY);
    }
    ...
}

50C {
    _bErrorCode[ _bLevel--] = READY;
    return(READY);
}

50D {
}

```

Fig. 6

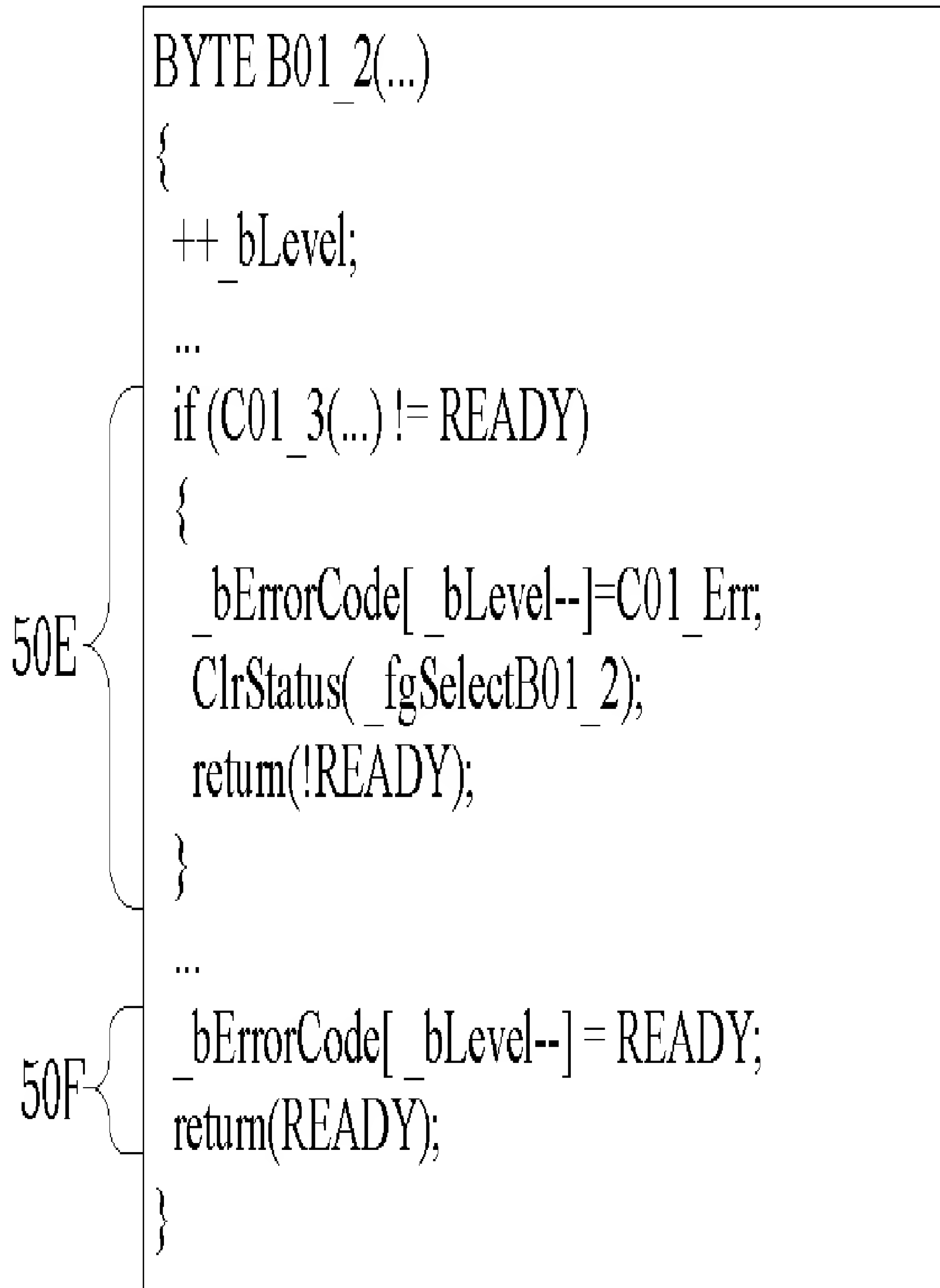


Fig. 7

```
BYTE C01_3(...)  
{  
    ...  
    D01_4(...);  
    ...  
    SetStatus(_fgSelectB01_2);  
    ...  
}
```

Fig. 8



```
BYTE D01_4(...)  
{  
    ...  
    E01_5(...);  
    ...  
    E02_5(...);  
    ...  
}
```

Fig. 9

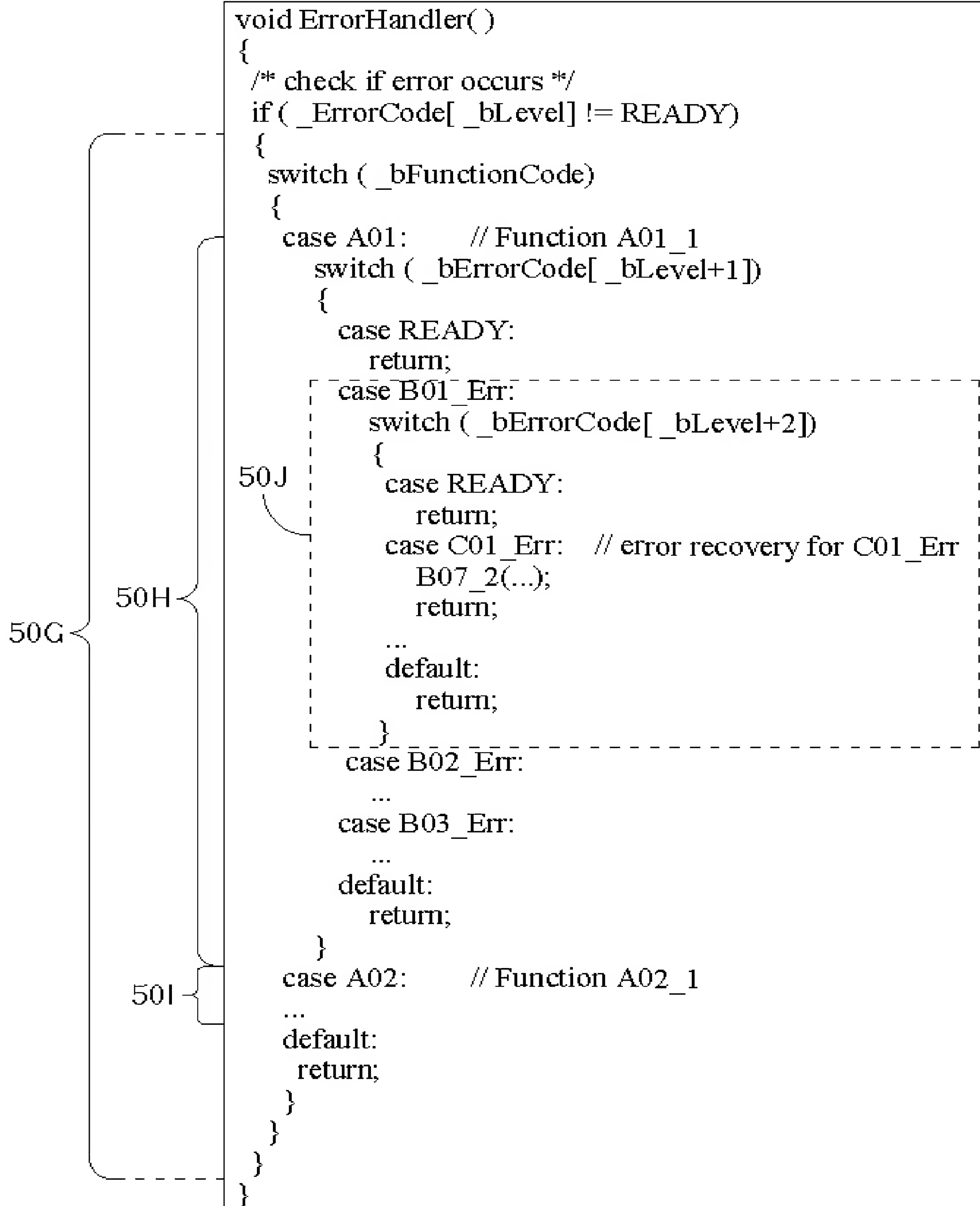


Fig. 10

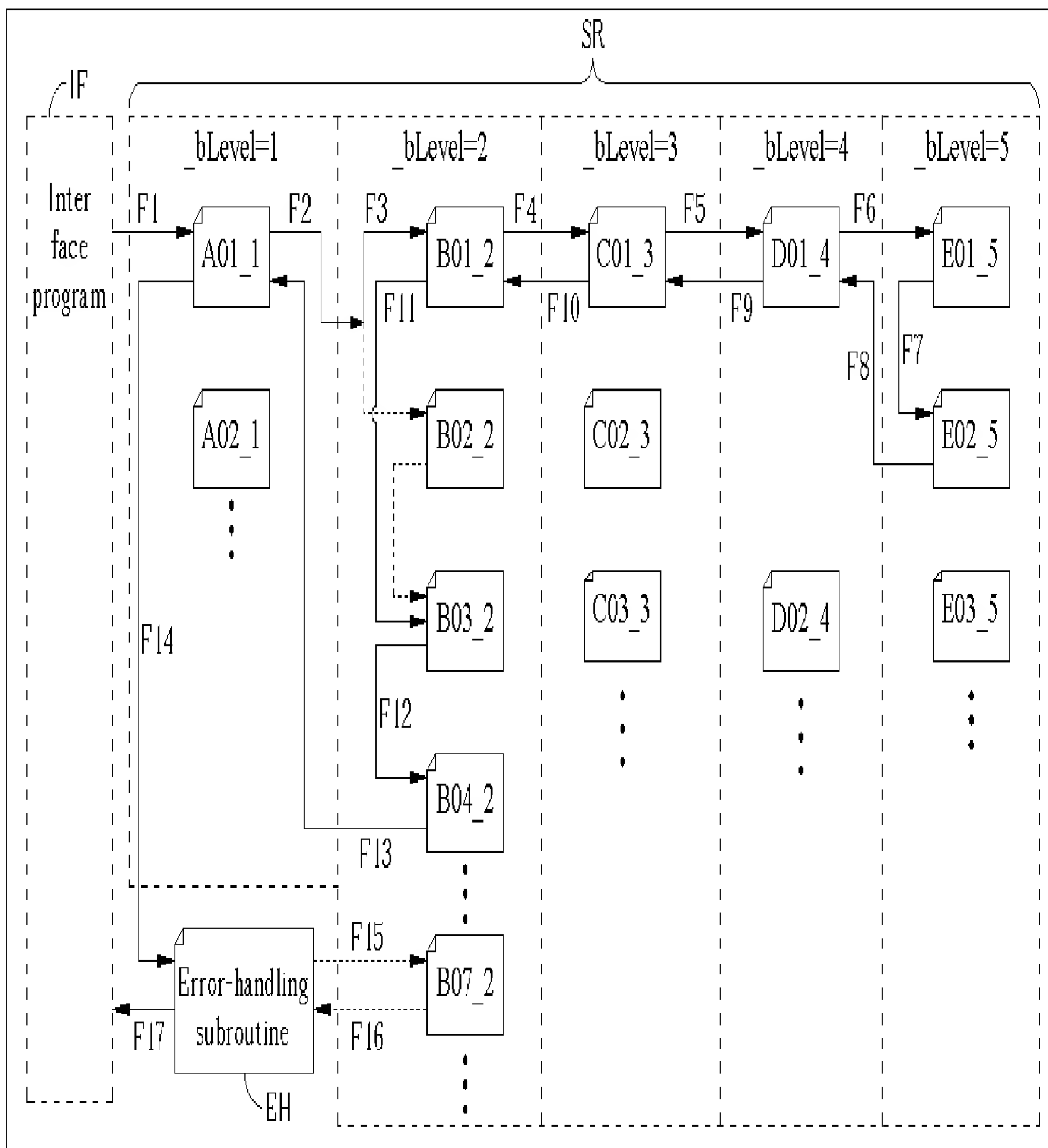


Fig. 11

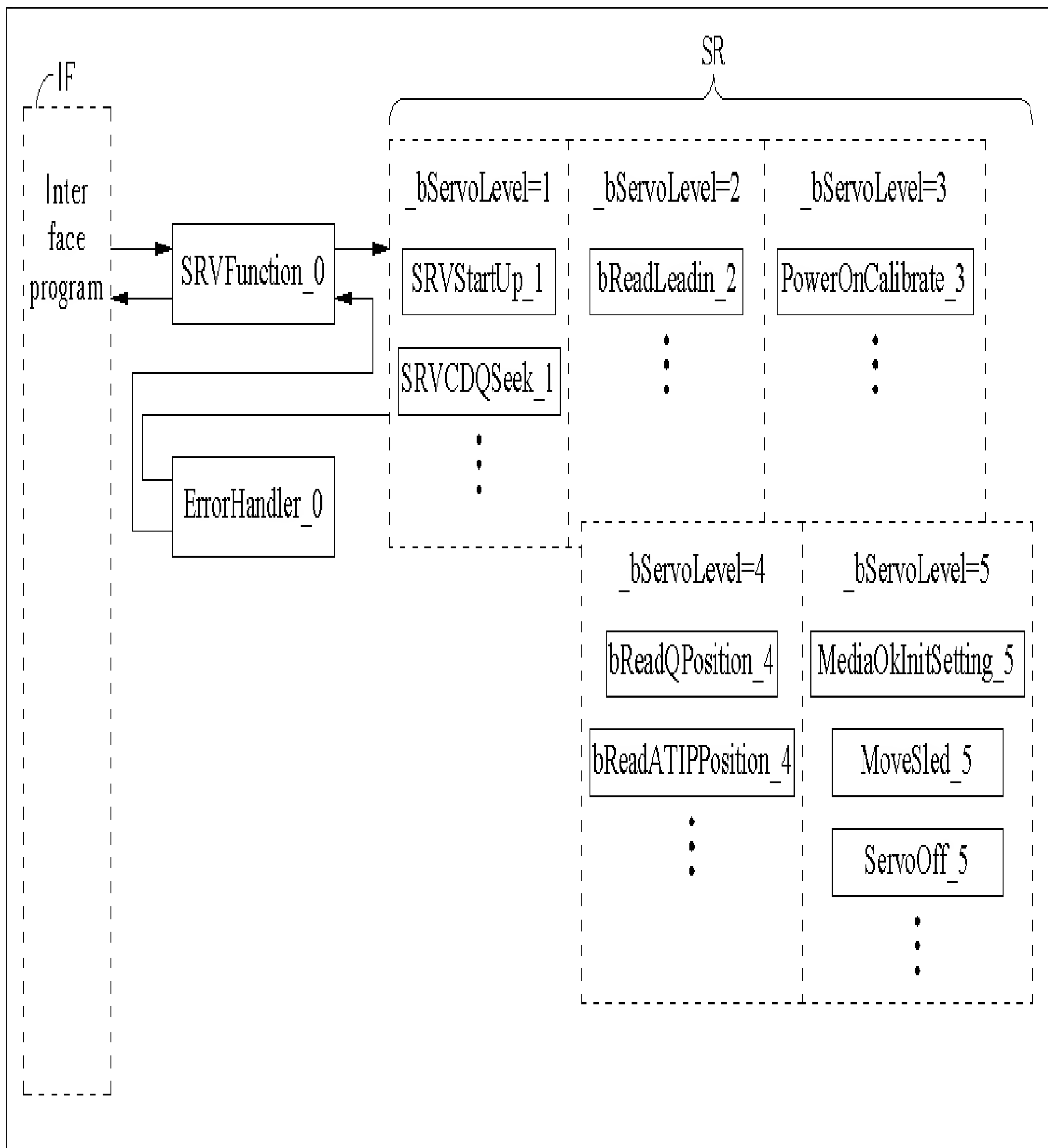


Fig. 12

```

#define ENTRY_LEVEL  0
#define MAX_ERR_CNT  3
#define RET(x)      {
                    _bErrorCode[ _bServoLevel--] = x ; \
                    return; \
                }
#define RET1(x)     {
                    _bErrorCode[ _bServoLevel--] = x ; \
                    return(x); \
                }
#define ChkStatus(x) x
#define SetStatus(x)  x = 1;
#define ClrStatus(x)  x = 0;

//-----
void SRVFunction_0(BYTE bFuncName )
{
    _bServoLevel = ENTRY_LEVEL;
    _bErrCnt = 0;
    _bErrorCode[ _bServoLevel] = bFuncName;

    do
    {
        switch (bFuncName)
        {
            case START_UP:
                SRVStartUp_1();
                break;

            case CD_Q_SEEK:
                SRVCDQSeek_1();
                break;

            default:
                break;
        }
        ErrorHandler_0();

    } while( _bErrorCode[0]!=EXIT_SRVFUNCTION);
}

```

Fig. 13

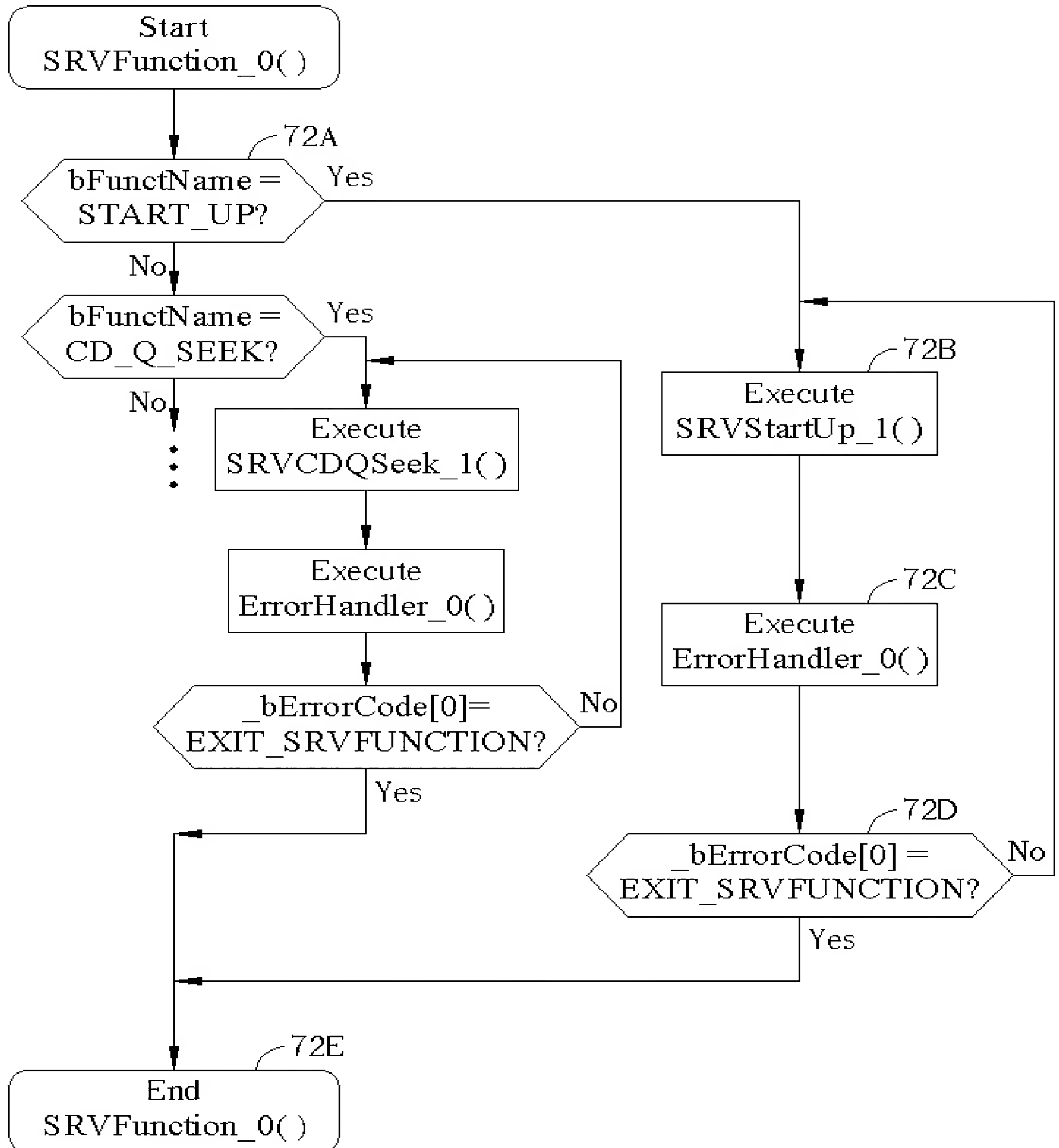


Fig. 14

```

void SRVStartUp_1( )
{
    _bServoLevel++;

    if (ChkStatus( _fgKEjtPressed )) RET( TRAY_EJECT );

    if (!ChkStatus( _fgPowerOnInit ))
    {
        PowerOnCalibrate_3( );
        MoveSled_5(TO_INNER, MOVE_SLED_HOME);
    }

    CheckMotorStop_5( );
    ...
    if(!_fgATIP)
    {
        if( bReadQPosition_4( ) ) RET( bReadQPosition_Err );
    }
    else
    {
        if( bReadATIPPosition_4( ) ) RET( bReadATIPPosition_Err );
        if( bReadLeadin_2( ) ) RET( bReadLeadin_Err );
    }

    MediaOkInitSetting_5( );

    RET( READY );
}

```

52A {

52B →

Fig. 15

```

void ErrorHandler_0( )
{
    switch ( _bErrorCode[0])
    {
        case START_UP:
            switch ( _bErrorCode[1])
            {
                case READY:
                    _bErrorCode[0] = EXIT_SRVFUNCTION;
                    _bPlayerStatus = READY;
                    return;

                case TRAY_EJECT:
                    _bErrorCode[0] = EXIT_SRVFUNCTION;
                    _bPlayerStatus = TRAY_EJECT;
                    return;

                case bReadQPosition_Err:
                    if( _fgDiskIsDVD)
                    {
                        _bMediaType=CDROM_DISC;
                    }
                    ServoOff_5();
                    _bPlayerStatus = _bErrorCode[2];
                    _bErrCnt++;
                    break;

                case bReadATIPPosition_Err:
                    ServoOff_5();
                    _bPlayerStatus = _bErrorCode[2];
                    _bErrCnt++;
                    break;
            }
        }
    }
}

```

Fig. 16



```

    case bReadLeadin_Err:
        switch( _bErrorCode[2])
        {
            case bSeekATIP_Err:
                _bPlayerStatus = _bErrorCode[3];
                switch( _bErrorCode[3])
                {
                    case FOCUS_ERROR:
                        ...
                    case READATIP_ERROR:
                        ...
                }
                break;

            case ReadLeadinInfo_Err:
                ...
        }
        break;

    default:
        break;
}
case CD_Q_SEEK
...
default:
    _bErrorCode[0] = EXIT_SRVFUNCTION;
    _bPlayerStatus = ILLEGAL_COMMAND;
    break;
}
if( _bErrCnt >= MAX_ERR_CNT )
{
    _bErrorCode[0] = EXIT_SRVFUNCTION;
}
}

```

54 {

Fig. 17

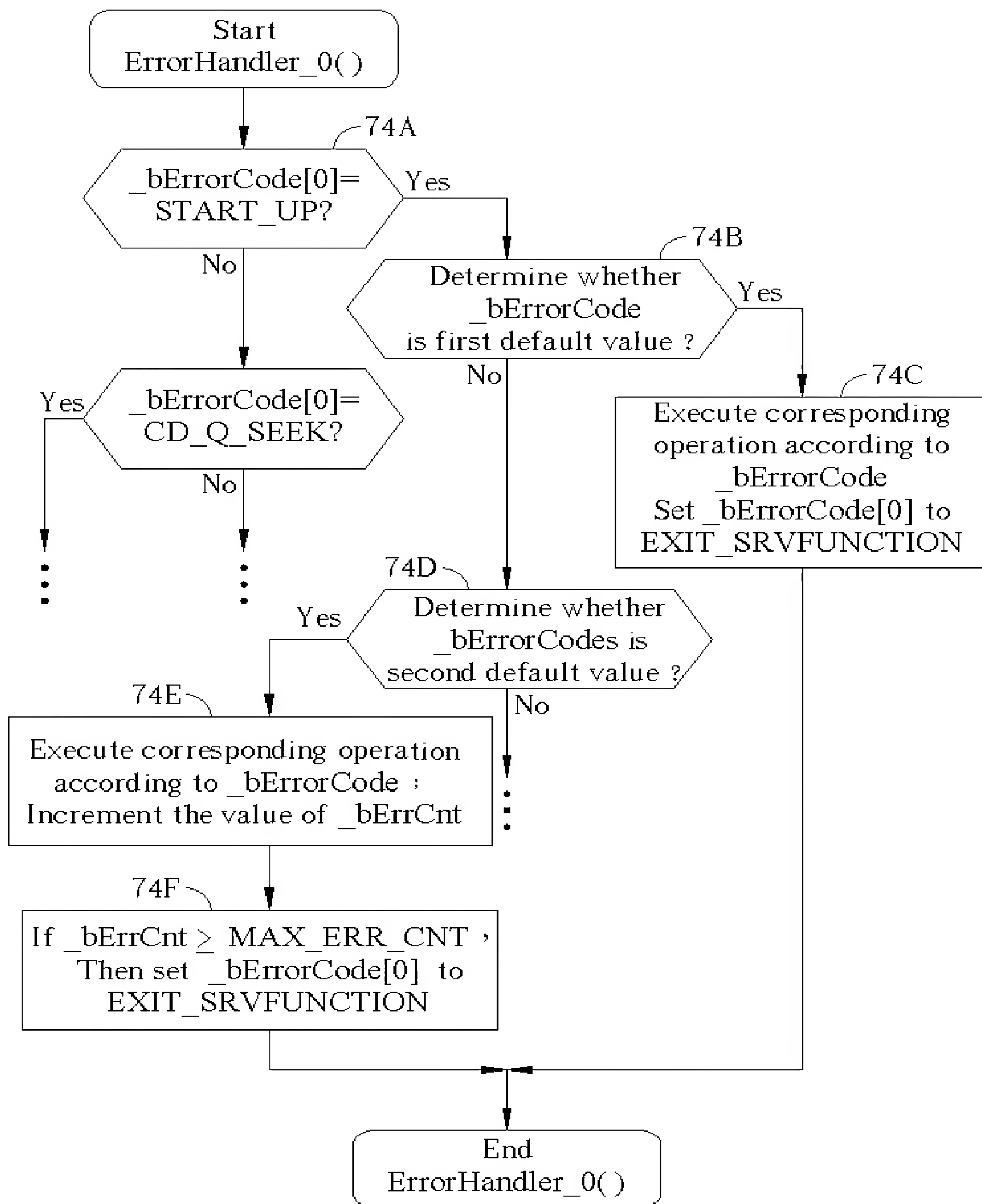


Fig. 18